DE19753288

Publication Title:

Information transmission method for fault-tolerant real-time computer system

Abstract:

The method involves using one or more broadcast communications channels (101,102) between the node processors (110,..150), each of which has an autonomous communication control device, coupled to the communications channels, providing access to the latter in synchronised time slots. The quitting of the information provided by a transmitting node processor is indicated by 2 test sum evaluations of the membership vector, ensuring that the information is correctly received by each reception node processor.

---

(19) BUNDESREPUBLIK DEUTSCHLAND

DEUTSCHES PATENTAMT

(12) **Offenlegungsschrift**

(10) **DE 197 53 288 A 1**

(51) Int. Cl.[6]:
G 06 F 15/163
G 06 F 11/00

(21) Aktenzeichen: 197 53 288.8
(22) Anmeldetag: 1. 12. 97
(43) Offenlegungstag: 4. 6. 98

DE 197 53 288 A 1

(30) Unionspriorität:
2104/96 03. 12. 96 AT

(71) Anmelder:
FTS Computertechnik Ges.m.b.H.,
Baden-Siegenfeld, AT

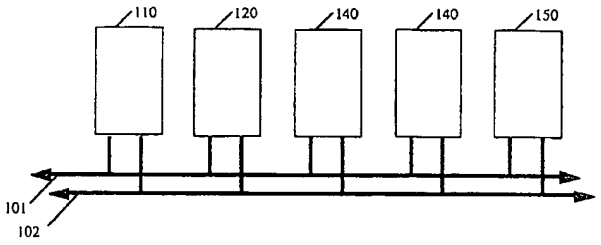
(74) Vertreter:
Lewald, D., Dipl.-Ing., Pat.-Anw., 80331 München

(72) Erfinder:
Kopetz, Hermann, Prof. Dr., Baden-Siegenfeld, AT

Die folgenden Angaben sind den vom Anmelder eingereichten Unterlagen entnommen

(54) Effizientes Quittungsverfahren in einem verteilten zeitgesteuerten Echtzeitsystem

(57) Quittungsverfahren in einem zeitgesteuerten verteilten fehlertoleranten Echtzeitcomputersystem, bei dem die Entscheidung, ob die Nachricht eines Senders bei den Empfängern richtig angekommen ist, aus der Auswertung der impliziten in der Prüfsumme enthaltenen Membershipinformation erfolgt. Wenn der Nachfolger des Senders den Sender als fehlerhaft beurteilt, so entscheidet der Nachfolger des Nachfolgers des Senders, ob der Sender oder der Nachfolger des Senders fehlerhaft ist. Kommunikationskontrolleinheit, die dieses Verfahren in Hardware umsetzt.



DE 197 53 288 A 1

Beschreibung

LITERATUR

Zitierte Patente:

US Patent:
4,866,606, 12. 9. 1989, Kopetz, H.

Europäisches Patent:
0 658 257, 18. 12. 96, Kopetz, H.

Internationale Patentanmeldung:
PCT/AT 93/00138, 2. 9. 1993, Kopetz, H.

Andere Veröffentlichungen:
Kopetz, H. und Ochsenreiter, W.; Clock Synchronization in Distributed Real-Time Systems, IEEE Transactions on Computers, vol. C-36, pp. 933–940, August 1987
Kopetz, H., & Gruensteidl, G. (1993). TTP-A Time-Triggered Protocol for Fault-Tolerant Real-Time Systems. Proc. 23rd IEEE International Symposium on Fault-Tolerant Computing (FTCS-23). Toulouse, France. IEEE Press. (pp. 524–532), appeared also in a revised version in IEEE Computer. Vol. 24 (1). (pp. 14–22).
SAE Handbook 1992, Vol. 2, pp. 20.301–20.302, Society of Automotive Engineers, 400 Commonwealth Drive, Warrendale, Pa, USA, 1992

TECHNISCHES UMFELD

Diese Erfindung betrifft ein zeitgesteuertes Kommunikationsverfahren und eine zeitgesteuerte Kommunikationskontrolleinheit zur Übertragung von Nachrichten in einem verteilten fehlertoleranten Echtzeitcomputersystem.

HINTERGRUND DIESER ERFINDUNG

Sicherheitskritische technische Anwendungen, d.s. Anwendungen wo ein Fehler im Zeitbereich zu einer Katastrophe führen kann, werden zunehmend von verteilten Computersystemen geführt.

In einem verteilten sicherheitskritischen Echtzeitcomputersystem, bestehend aus einer Anzahl von Knotenrechnern und einem Echtzeitkommunikationssystem, muß der Ausfall eines Knotenrechners mit geringer Verzögerung erkannt werden. Im Kern einer solchen Computerarchitektur befindet sich ein fehlertolerantes Echtzeitkommunikationssystem zum vorhersehbar schnellen und sicheren Austausch von Nachrichten. In einem Echtzeitkommunikationssystem soll die Länge der Nachrichten so kurz wie nur möglich sein, um eine schnelle Reaktion des Systems auch bei geringer verfügbarer Bandbreite zu ermöglichen.

Bekannten Methoden der Nachrichtenübertragung in verteilten Echtzeitsystemen sind J1850, CAN und andere. Eine Gegenüberstellung dieser Methoden findet sich im 1992 SAE Handbook, Vol., pp. 20.301–20.302, Society of Automotive Engineers, 400 Commonwealth Drive, Warrendale, Pa, USA. Keine dieser Methoden bietet spezielle Dienste an, die den Bau von fehlertoleranten Rechnersystemen vereinfachen.

Die Implementierung der Fehlertoleranz in verteilten Computersystemen wird vereinfacht, wenn das Kommunikationssystem folgende Leistungen erbringt: Rechtzeitige Übertragung von Nachrichten auch im Fehlerfall, fehlertolerante Uhrensynchronisation, einen Membershipdienst, um den Ausfall eines Knotens konsistent erkennen zu können, und einen Rekonfigurationsdienst, um ausgefallene Knoten dynamisch ersetzen zu können.

Ein Protokoll, das diese Anforderungen erfüllt ist im zitierten Europäischen Patent 0 658 257 v. 18.12.96 sowie in der Internationalen Patenanmeldung PCT/AT 93/00 138 beschrieben. Das Protokoll ist unter dem Namen "Time-Triggered Protokoll (TTP) bekannt geworden. TTP verwendet ein Verfahren zur fehlertoleranten Uhrensynchronisation, das im US Patent: 4,866,606 offengelegt wurde.

In TTP werden replizierte fail-silent Knotenrechner zu einer fehlertoleranten Einheit (Fault-Tolerant Unit-FTU) zusammengefaßt. Solange ein Knotenrechner einer FTU funktioniert, werden die Dienste der FTU im Zeit- und Wertebereich erbracht. Die Membership in TTP bezieht sich auf die Funktion der FTU.

Wenn nun in einer Anwendung replizierte und nicht replizierte Knotenrechner nebeneinander eingesetzt werden, so ergeben sich beim TTP folgende Nachteile:

(i) Die nicht replizierten Knotenrechner beanspruchen die volle Bandbreite einer FTU, was zu einem Verlust von 50% der Bandbreite führt.

(ii) Wenn alle Nachrichten eines nicht replizierten Knotenrechners gestört werden verliert der Knotenrechner seine Membership und kann aufgrund des impliziten Quittungsmechanismus von TTP die nächste Nachricht nicht richtig empfangen.

Diese Nachteile werden durch die vorliegende Erfindung behoben.

ZUSAMMENFASSUNG

Es ist das wesentliche Ziel der vorliegenden Erfindung die Dateneffizienz, Robustheit und Flexibilität von zeitgesteu-

erten Protokollen, wie z. B. von TTP, durch ein neues implizites Quittungsverfahren wesentlich zu verbessern.

Dieses Ziel wird dadurch erreicht, daß sich die Membership auf die Funktion der Knotenrechner (und nicht auf die Funktion der FTU) bezieht und daß auf die in einem zeitgesteuerten Protokoll vorgesehenen Quittungsbits im Nachrichtenkopf verzichtet wird. Entsprechend dieser Erfindung erfolgt die Quittung der Nachrichten implizit über ein erweitertes Auswerteverfahren des Membershipvektors.

KURZE BESCHREIBUNG DER ABBILDUNGEN

Das vorab beschriebene Ziel und andere neue Eigenschaften der vorliegenden Erfindung werden in den angeführten Abbildungen erläutert.

Fig. 1 zeigt die Struktur eines verteilten Computersystems mit fünf Knotenrechnern, die über duplizierte Kommunikationskanäle verbunden sind.

Fig. 2 zeigt die Struktur eines Knotenrechners, bestehend aus einer Kommunikationskontrolleinheit und einem Host Computer, die über das Communication Network Interface (CNI) kommunizieren.

Fig. 3 zeigt die Datenstruktur, die den Sendezeitpunkt von Nachrichten vorgibt.

Fig. 4 zeigt die Folge von Nachrichten, wie sie auf den beiden Kommunikationskanälen gesendet werden.

Fig. 5 zeigt den Aufbau des Membershipvektors.

Fig. 6 zeigt den Aufbau einer Nachricht.

Fig. 7 zeigt die Felder, die bei der CRC Berechnung normaler Nachrichten berücksichtigt werden.

BESCHREIBUNG EINER REALISIERUNG

Im folgenden Abschnitt wird eine Realisierung des neuen Verfahrens an einem Beispiel mit fünf Knotenrechnern, die über zwei replizierte Kommunikationskanäle verbunden sind, gezeigt. Die Objekte in den Abbildungen sind so numeriert, daß die erste der dreistelligen Objektziffern immer die Bildnummer angibt.

Fig. 1 zeigt ein System von fünf Knotenrechnern **110**, **120**, **130**, **140**, und 150. Ein Knotenrechner bildet eine austauschbare Einheit (SRU, smallest replaceable unit). Jeder Knotenrechner ist über zwei Anschlüsse mit den replizierten Kommunikationskanälen **101** und **102** verbunden. Die beiden Kommunikationskanäle unterstützen Broadcast, d. h., jede Nachricht kann im fehlerfreien Fall von allen Knotenrechnern empfangen werden.

Fig. 2 zeigt den inneren Aufbau eines Knotenrechners. Er besteht aus zwei Subsystemen der Kommunikationskontrolleinheit **210**, die mit den replizierten Kommunikationskanälen **201** und **202** verbunden ist, und den Host Computer **220**, auf dem die Anwendungsprogramme des Knotenrechners ausgeführt werden. Diese beiden Subsystemen sind über das Communication Network Interface (CNI) **241** und eine Signalleitung **242** verbunden. Das CNI besteht aus einem Speicher (Dual Ported RAM, DPRAM) **241** auf das beide Subsysteme zugreifen können. Die beiden Subsysteme tauschen über diesen gemeinsamen Speicher **241** die Kommunikationsdaten aus. Die Signalleitung **242** dient zur Übertragung der synchronisierten Zeitsignale. Diese Signalleitung ist im angeführten US Patent 4,866,606 genau beschrieben. Die Kommunikationskontrolleinheit **210** verfügt über eine Kommunikationskontrolleinheit **211** und eine Datenstruktur **212** die angibt, zu welchen Zeitpunkten Nachrichten gesendet und empfangen werden müssen. Die Datenstruktur **212** wird als Message Descriptor List (MEDL) bezeichnet.

Die Datenstruktur **212** hat die Form einer Tabelle wie in **Fig.** 3 dargestellt. In der ersten Spalte dieser Tabelle 301 sind die Zeitpunkte der global synchronisierten Zeit eingetragen zu denen Nachrichten empfangen oder gesendet werden müssen. Die nächste Spalte **302** gibt an, von welcher Adresse des DPRAM **241** die Nachrichten gelesen werden oder wohin im DPRAM **241** die Nachrichten geschrieben werden. Die dritte Spalte **303** enthält weitere Attribute der Nachrichten, wie die Nachrichtenlänge und ob es sich bei der Nachricht um eine Eingabenachricht oder eine Ausgabenachricht handelt.

Fig. 4 zeigt den zeitlichen Verlauf einer Senderunde auf den beiden Kommunikationskanälen **101** und **102**. Die globale Zeit schreitet von links nach rechts. Die Knotenrechner senden ihre Nachrichten nach dem bekannten zyklischen Zeitscheibenverfahren (Time-Division Multiple Access TDMA). Jeder Knotenrechner sendet in jeder TDMA Runde gleichzeitig eine Nachricht auf jedem der beiden replizierten Kanäle, dem Kanal **401** und dem Kanal **402**. Zwischen dem Senden von zwei aufeinanderfolgenden Nachrichten erfolgt eine Pause, die als Interframe Gap (IFG) (**411**, **421**, **431**, **441**, **451**) bezeichnet wird. Im Zeitintervall **410** sendet Knotenrechner **110** mit dem folgenden IFG **411**, anschließend sendet Knotenrechner **120** während dem Zeitintervall **420** mit dem IFG **421**, dann Knotenrechner **130** während dem Zeitintervall **430** mit dem IFG **431**, dann Knotenrechner **140** während dem Zeitintervall **440** mit dem IFG **441** und schließlich Knotenrechner **150** während dem Zeitintervall **450** mit dem IFG **451**.

Fig. 5 zeigt den Membershipvektor eines Knotenrechners. Der Membershipvektor ist ein Bitvektor, dessen Länge der Anzahl der Knotenrechner entspricht. Jedem Knotenrechner ist eine Bitposition im Membershipvektor zugeordnet. Im gegebenen Beispiel hat der Membershipvektor eine Länge von 5 Bits. Bit **510** ist dem Knotenrechner **110**, Bit **520** ist dem Knotenrechner **120**, Bit **530** ist dem Knotenrechner **130**. Bit **540** ist dem Knotenrechner **140** und Bit **550** ist dem Knotenrechner **150** zugeordnet. Wenn ein Knotenrechner funktioniert, so ist das entsprechende Bit "WAHR", wenn er nicht funktioniert, so ist das entsprechende Bit "FALSCH". Jeder Knotenrechner verfügt über eine eigene Sicht des Membershipvektors. Diese verschiedene Sichten werden durch das Protokoll konsistent gehalten.

Fig. 6 zeigt den Aufbau einer Nachricht. Eine Nachricht besteht aus drei Feldern, dem Header **601**, dem Datenfeld variabler Länge **602**, und dem Prüfsummenfeld **603**. Der Header **601** besteht aus 4 Bits, dem Initialisierungs/Normal (I/N) Bit **611** das angibt ob es sich bei der Nachricht um eine Initialisierungsnachricht oder eine normale Nachricht handelt, und drei Modewechselbits (**612**, **613**, **614**) die angeben, ob nach dem Empfang der Nachricht ein Betriebsartenwechsel durchgeführt werden soll.

Fig. 7 zeigt die Datenfelder, die bei der CRC Berechnung berücksichtigt werden. Entsprechend dem Europäischen Patent 0 658 257 wird die Prüfsumme jeder normalen Nachricht über die Verkettung des Headers **701**, des Datenfeldes **702**,

und dem inneren Zustand des Senders **703** gebildet. Der innere Zustand des Senders besteht aus der globalen Zeit des Senders, der gegenwärtigen Position in der MEDL **212** und dem Membershipvektor (**Fig. 5**). Der Empfänger überprüft das CRC unter Berücksichtigung des inneren Zustandes des Empfängers und kann daher erkennen, ob der Sender und Empfänger den gleichen Membershipvektor besitzen.

Nachdem die einzelnen Bausteine beschrieben wurden, wird nun die Realisierung der Erfindung anhand eines Ablaufs entsprechend den **Abb.** 1–7 erklärt. Zuerst wird ein fehlerfreier Ablauf der Kommunikation beschrieben.

Im IFG **401** bereitet sich der Knotenrechner **110** auf das Senden der Nachricht **410** vor. Er setzt sein Membership Bit **510** auf "WAHR" und berechnet die Prüfsumme unter Berücksichtigung seines aktuellen inneren Zustands. Dann beginnt Knotenrechner **110** mit dem Senden der Nachricht auf beiden Kommunikationskanälen **101** und **102**. Im IFG **411** empfängt Knotenrechner **120** die beiden Nachrichten von Knotenrechner **110**. Knotenrechner **120** verkettet die empfangene Nachricht **410** mit seinem aktualisierten Membershipvektor und errechnet die Prüfsumme der Nachricht **410**. Der Membershippunkt eines Knotenrechners ist das IFG nach dem Senden der Nachricht, also IFG **411** für den Knotenrechner **110**. Wenn die Prüfsumme bei mindestens einer der beiden Nachrichten richtig ist, so schließt Knotenrechner **120**, daß Knotenrechner **110** im IFG **411** funktioniert hat und der innere Zustand **703** von Knotenrechner **110** mit dem inneren Zustand **703** von Knotenrechner **120** übereinstimmt. Knotenrechner **120** setzt sein Membershipbit **510** von Knotenrechner **110** auf "WAHR". Knotenrechner **120** bereitet sich während IFG **411** auf das Senden seiner Nachricht vor. Er setzt sein Membershipbit **520** auf "WAHR" und berechnet die Nachrichtenprüfsumme mit seinem neuen aktuellen inneren Zustand. Dann sendet Knotenrechner **120** im Zeitintervall **420** seine Nachricht. Im IFG **421** empfängt Knotenrechner **110** die Nachricht vom Knotenrechner **120**. Knotenrechner **110** errechnet sich die Prüfsumme der Nachricht **420** unter der Annahme, daß er zu seinem Membership Punkt aus der Sicht von Knotenrechner **120** funktioniert hat (Auswertung 1: AW1). Stimmt die Prüfsumme so ist die Annahme bestätigt und die Quittung der Nachricht **410** von Knotenrechner **110** durch den Nachfolger **120** positiv ausgefallen. Im fehlerfreien Fall wird dieser Algorithmus nach jedem Sendevorgang in gleicher Weise ausgeführt. Der Nachfolger von Knotenrechner **150** ist Knotenrechner **110**, entsprechend dem Ablauf des TDMA Verfahrens.

Der folgende Abschnitt beschreibt die Funktion des innovativen Quittungsverfahrens im Fehlerfall. Es wird hier angenommen, daß innerhalb einer TDMA Runde einer der folgenden Fehler auftreten kann:

(i) Der sendende Knotenrechner macht einen Fehler.
(ii) Beide Nachrichten von einem Knotenrechner werden beim Transport gestört.
(iii) Der empfangene Knotenrechner macht einen Fehler.

Wenn im vorab beschriebenen Fall die Überprüfung der Nachricht **420** durch Knotenrechner **110** negativ ausfällt, so setzt Knotenrechner **110** vor einer zweiten Berechnung der Prüfsumme (die parallel oder sequentiell zur ersten Berechnung ausgeführt werden kann) sein Membershipbit **510** auf "FALSCH" (Auswertung 2: AW2). Wenn auch diese zweite Überprüfung durch Knotenrechner **110** keine richtige Prüfsumme ergibt, so nimmt Knotenrechner **110** an, daß beide Nachrichten vom Knotenrechner **120** gestört wurden. Knotenrechner **110** nimmt Knotenrechner **120** aus der Membership indem er das Membershipbit **520** von Knotenrechner **120** auf "FALSCH" setzt. Von nun an ist Knotenrechner **130** der direkte Nachfolger von Knotenrechner **110**.

Stimmt die zweite Überprüfung, so hat sich die Hypothese bestätigt daß Knotenrechner **120** annimmt, Knotenrechner **110** sei fehlerhaft. Diese Situation kann zwei Gründe haben

(i) Die Nachrichten von Knotenrechner **110** wurden auch von anderen Knotenrechnern nicht richtig empfangen, d. h. Knotenrechner **110** ist fehlerhaft.
(ii) Knotenrechner **120** ist fehlerhaft.

Die Entscheidung welche dieser Alternativen stimmt, wird Knotenrechner **130** übertragen.

Wenn nun die Nachricht **430** von Knotenrechner **130** bei Knotenrechner **110** eintrifft, so berechnet Knotenrechner **110** zwei Prüfsummen mit folgenden Annahmen:

(i) Knotenrechner **110** setzt das Membershipbit **510** auf "FALSCH" und Membershipbit **520** auf "WAHR". Das bedeutet, daß Knotenrechner **110** annimmt er wird als fehlerhaft und Knotenrechner **120** als richtig angesehen (Auswertung 3: AW3).
(ii) Knotenrechner **110** setzt Membershipbit **510** auf "WAHR" und Membershipbit **520** auf "FALSCH". Das bedeutet, daß Knotenrechner **110** annimmt er wird als richtig und Knotenrechner **120** als fehlerhaft angesehen (Auswertung 4: AW4).

Wenn die Prüfsumme unter der Annahme (i) stimmt, dann hat Knotenrechner **130** entschieden daß Knotenrechner **120** recht hat. Knotenrechner **110** setzt sein Membershipbit **510** auf "FALSCH" und das Membershipbit **520** von Knotenrechner **120** auf "WAHR".

Wenn die Prüfsumme unter der Annahme (ii) stimmt, dann hat Knotenrechner **130** entschieden, daß Knotenrechner **110** recht hat. Knotenrechner **110** setzt sein Membershipbit **510** auf "WAHR" und das Membershipbit **520** von Knotenrechner **120** auf "FALSCH".

Wenn keine dieser Annahmen zu einer richtigen Prüfsumme führt, so nimmt Knotenrechner **110** an daß ein Doppelfehler vorliegt. Um die fail-silent Eigenschaft zu wahren, setzt er sein Membershipbit **510** auf "FALSCH".

Insgesamt werden vom ursprünglichen Sender **110** zwei CRC Auswertungen der Nachricht des Nachfolgers **120** und zwei CRC Auswertungen der Nachricht des Nachfolgers des Nachfolgers **130** vorgenommen:

Auswertung	Nachricht kommt vom	Membership des urspr.Senders	Membership des Nachfolgers
AW 1	Nachfolger	"WAHR"	"WAHR"
AW 2	Nachfolger	"FALSCH"	"WAHR"
AW 3	Nachfolger des Nachfolgers	"FALSCH"	"WAHR"
AW 4	Nachfolger des Nachfolgers	"WAHR"	"FALSCH"

Es ergeben sich somit folgende Fallunterscheidungen beim ursprünglichen Sender einer Nachricht:

	AW 1	AW 2	AW 3	AW 4	
Fall 1	"WAHR"	"FALSCH"	don't care	don't care	Normalfall, Nachricht OK
Fall 2	"FALSCH"	"FALSCH"	don't care	don't care	Übertragungsfehler, Nachfolger verliert Membership
Fall 3	"FALSCH"	"WAHR"	"WAHR"	"FALSCH"	ursp. Sender fehlerhaft
Fall 4	"FALSCH"	"WAHR"	"FALSCH"	"WAHR"	Nachfolger fehlerhaft
alle anderen Fälle					Doppelfehler

Gegenüber dem im Europäischen Patent 0 658 257 veröffentlichten Protokoll bietet die vorliegende Erfindung die folgenden wirtschaftlichen Vorteile

(i) Die Knotenrechnermembership ermöglicht die Bildung von unterschiedlichen FTU Konfigurationen, z. B., eine FTU mit zwei failsilent Rechnern, wie in TTP, oder eine FTU mit drei Rechnern (TMR-Triple Modular Redundancy).
Da vorgesehen ist, das vorliegende Protokoll in einem VLSI Chip zu implementieren wird das Einsatzgebiet eines solchen Chips wesentlich erweitert.
(ii) Wenn in einem System ein nicht replizierter Knotenrechner eingesetzt wird, so ist auch nur eine einzige Sendezeitscheibe vorzusehen. In einem System mit zwei replizierten und vier nicht replizierten Knotenrechnern führt dies zu einer Dateneffizienzsteigerung von 33% gegenüber der FTU Membership.
(iii) Ein Knotenrechner, dessen Nachrichten gestört wurden und der deshalb seine Membership verliert, kann alle weiteren Nachrichten empfangen. Dies erleichtert die Reintegration des Knotenrechners.
(iv) Auf die Quittungsbits im Nachrichtenkopf kann verzichtet werden, was zu einer weiteren Erhöhung der Dateneffizienz führt. Nimmt man an, daß eine Nachricht eine Nutzdatenlänge von 12 Bit hat (entpricht einem Sensorwert), so wird allein durch den Wegfall der Quittungsbits die Dateneffizienz gegenüber dem TTP Protokoll um ca. 10% verbessert.

Abschließend sei festgehalten, daß diese Erfindung sich nicht auf die beschriebene Realisierung mit fünf Knotenrechnern und zwei Bussen beschränkt. Die beschriebene Auswertelogik zur Quittung der Nachrichten kann in einem hochintegrierten Mikrokontroller der alle in **Fig.** 2 gezeigten Funktionseinheiten auf einem einzigen Chip beinhaltet, in Software oder in Hardware realisiert werden.

Patentansprüche

1. Methode zur Übertragung von Nachrichten in einem zeitgesteuerten verteilten fehlertoleranten Computersystem, indem eine Vielzahl von Knotenrechnern über einen oder mehrere Broadcast Kommunikationskanäle verbunden sind, und wo jeder Knotenrechner über eine autonome Kommunikationskontrolleinheit mit den entsprechenden Anschlüssen an die Kommunikationskanäle verfügt und wo der Zugriff auf die Kommunikationskanäle entsprechend einem zyklischen Zeitscheibenverfahren erfolgt, und wo die Prüfsumme der Nachricht beim Sender über die zu sendende Nachricht verkettet mit dem inneren Zustand des Senders und beim Empfänger über die empfangene Nachricht verkettet mit dem inneren Zustand des Empfängers errechnet wird, **dadurch gekennzeichnet**, daß die Quittung der Nachricht eines sendenden Knotenrechners implizit aus dem Ergebnis von je zwei Prüfsummenauswertungen betreffend die Nachricht des Nachfolger des Senders und die Nachricht des Nachfolgers des Nachfolgers des Senders derart abgeleitet wird, daß zuerst im Membershipvektor angenommen wird der ursprüngliche Sender und der Nachfolger sind funktionsfähig was zu einer positiven Quittung der Nachricht führt, oder daß der ursprüngliche Sender annimmt, der Nachfolger betrachte den ursprünglichen Sender als fehlerhaft, was zu einer Verlagerung der endgültigen Entscheidung, wer funktionsfähig ist, an den Nachfolger des Nachfolgers führt, wobei die Auswertung der Prüfsumme des Nachfolgers des Nachfolgers entscheidet, ob der Nachfolger oder der ursprünglich sendende Knotenrechner fehlerhaft ist.
2. Kommunikationsmethode nach Anspruch 1, dadurch gekennzeichnet, daß die Nachfolgerelation dynamisch aus dem zum Zeitpunkt der Auswertung gültigen Zustand des Membershipvektors abgeleitet wird.
3. Kommunikationsmethode nach den Ansprüchen 1 oder 2, dadurch gekennzeichnet, daß die Nachricht auf dem Übertragungskanal aus einem Nachrichtenheader, bestehend aus einem I/N Bit und drei Betriebsartenänderungs-

bits, einem Datenfeld variabler Länge, und einem Prüfsummenfeld besteht.

4. Kommunikationskontrolleinheit zur Übertragung von Nachrichten in einem zeitgesteuerten verteilten fehlertoleranten Computersystem, indem eine Vielzahl von Knotenrechnern über einen oder mehrere Broadcast Kommunikationskanäle verbunden sind und wo jeder Knotenrechner über eine autonome Kommunikationskontrolleinheit mit den entsprechenden Anschlüssen an die Kommunikationskanäle verfügt und wo der Zugriff auf die Kommunikationskanäle entsprechend einem zyklischen Zeitscheibenverfahren erfolgt, und wo die Prüfsumme der Nachricht beim Sender über die zu sendende Nachricht verkettet mit dem inneren Zustand des Senders und beim Empfänger über die empfangene Nachricht verkettet mit dem inneren Zustand des Empfängers errechnet wird, dadurch gekennzeichnet, daß die Quittung der Nachricht eines sendenden Knotenrechners implizit aus dem Ergebnis von je zwei Prüfsummenauswertungen betreffend die Nachricht des dynamischen Nachfolger des Senders und die Nachricht des dynamischen Nachfolgers des Nachfolgers des Senders derart abgeleitet wird, daß zuerst im Membershipvektor angenommen wird der ursprüngliche Sender und der Nachfolger sind funktionsfähig was zu einer positiven Quittung der Nachricht führt, oder daß der ursprüngliche Sender annimmt, der dynamische Nachfolger betrachte den ursprünglichen Sender als fehlerhaft was zu einer Verlagerung der endgültigen Entscheidung wer funktionsfähig ist an den Nachfolger des Nachfolgers führt, wobei die Auswertung der Prüfsumme des Nachfolgers des Nachfolgers entscheidet, ob der Nachfolger oder der ursprünglich sendende Knotenrechner fehlerhaft ist.

Hierzu 2 Seite(n) Zeichnungen

Nummer: DE 197 53 288 A1
Int. Cl.[6]: G 06 F 15/163
Offenlegungstag: 4. Juni 1998

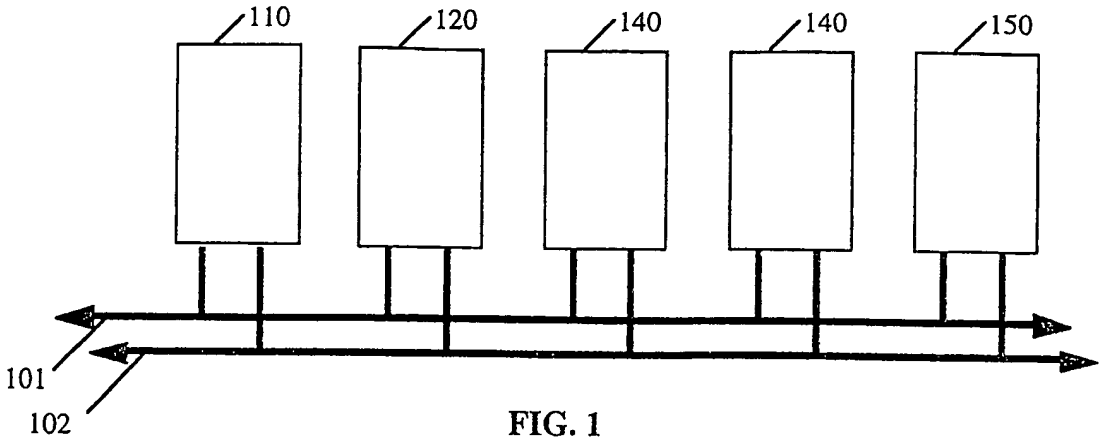


FIG. 1

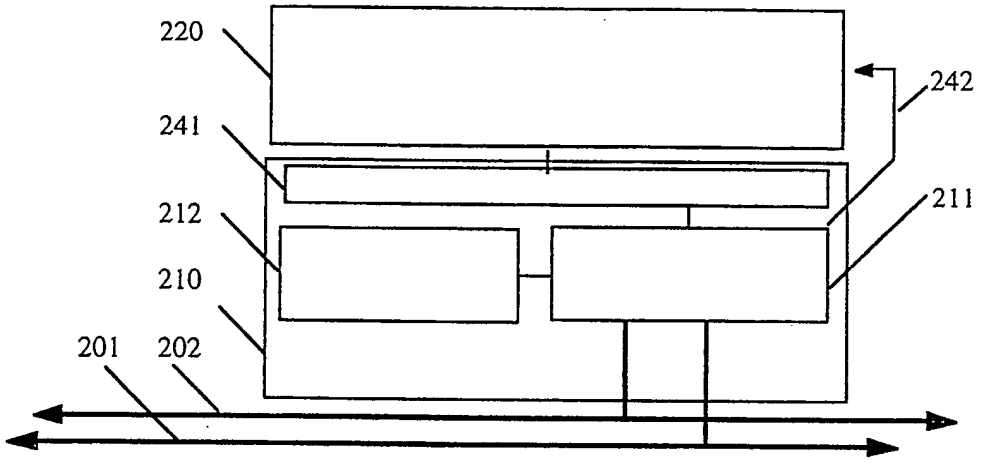


FIG. 2

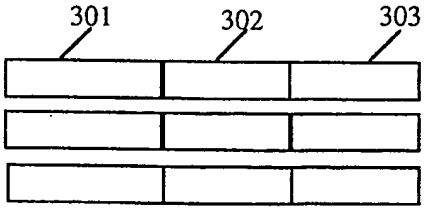


FIG. 3

Nummer: DE 197 53 288 A1
Int. Cl.[6]: G 06 F 15/163
Offenlegungstag: 4. Juni 1998

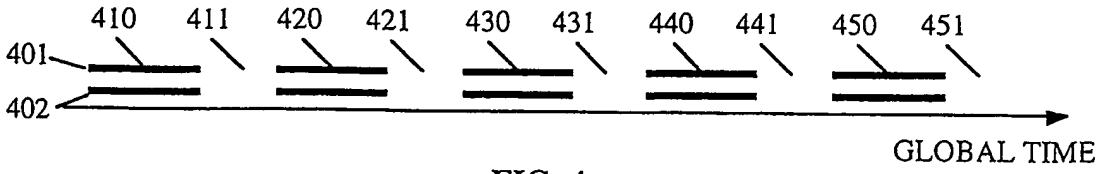


FIG. 4

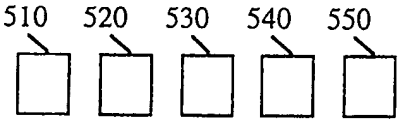


FIG. 5

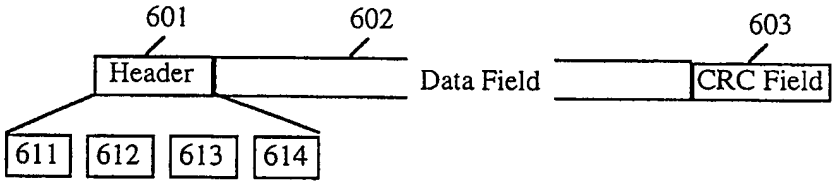


FIG. 6

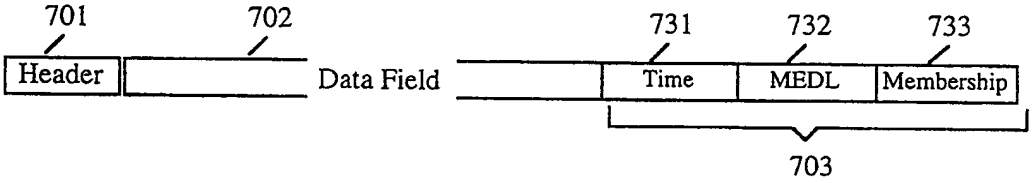


FIG. 7